IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application Inventor(s): Patrick Chiu et al. Appln. No.: 10/636,044 Confirm. No.: 9973 Filed: August 8, 2003 Title: PEER TO PEER GESTURE BASED MODULAR PRESENTATION SYSTEM	PATENT APPLICATION Art Unit: 2179 Examiner: Nicholas Augustine Atty. Docket No.: FXPL-01060US0 **Customer No. 23910**

DECLARATION OF INVENTORS UNDER 37 C.F.R. § 1.131

1. We the undersigned, Patrick Chiu, Qiong Liu and John Steven Boreczky, declare as follows:

2 We are employees of FX Palo Alto Laboratories and specialize in computer software research and development.

3. We are the inventors of the invention described and claimed in the above U.S. patent application.

4. For convenience, embodiments of this invention hereafter are referred to as the "ModSlideShow".

5. For convenience, Claim 1 of the above application is set out below with elements (a)-(c) identified as follows:

   A system for providing content in a modular presentation system, comprising:

   (a) a plurality of displays, wherein each display neighbors at least one other display and at least two of the plurality of displays are in visual proximity to each other;

   (b) an input device that receives input of a gesture to move a content from a first display of the plurality of displays; and

   (c) a processor, that interprets a direction to move the content from the first display based on the gesture, that specifies a second display to which the content is to be moved, based on the gesture and the position of the plurality of displays and that propagates the content of the first display to the second display.

6. A prototype of the "ModSlideShow" which for convenience we identify as "ModSlideShow1" was reduced to practice on or before April 18, 2002.

7. The "ModSlideShow1" prototype was implemented with a rear projector and touch screen plasma displays.

8. The "ModSlideShow1" prototype was implemented with laptops connected on a wireless network, where a location sensing system can detect the room location of a laptop.

9. The "ModSlideShow1" prototype integrated a digital compass with a laptop.

10. The "ModSlideShow1" prototype contained all the elements of Claim 1 as described and claimed in the above U.S. patent application.

11. The "ModSlideShow1" prototype comprises a plurality of displays, wherein each display neighbors at least one other display and at least two of the plurality of displays are in visual proximity to each other as claimed in Claim 1, element (a).

12. The "ModSlideShow1" prototype comprises an input device that receives input of a gesture to move a content from a first display of the plurality of displays as claimed in Claim 1, element (b).

13. The "ModSlideShow1" prototype comprises a processor that interprets a direction to move the content from the first display based on the gesture, that specifies a second display to which the content is to be moved, based on the gesture and the position of the plurality of displays and that propagates the content of the first display to the second display, as claimed in Claim 1, element (c).

14. For convenience, Claim 5 of the above application is set out below with elements (d)-(g) identified as follows:

A method of providing content in a modular presentation system having a plurality of displays, wherein at least two of the plurality of displays are, the method comprising:

(d) receiving input of a gesture to move a first content presented on a first display of the plurality of displays;

(e) interpreting a direction to move the content from the first display based on the gesture;

(f) specifying a second display to which the first content is to be moved based on the gesture and the relative position of the plurality of displays; and

(g) presenting the first content at the second display and a second content at the first display.

15. The "ModSlideShow1" prototype contained all the elements of Claim 5 as described and claimed in the above U.S. patent application.

16. The "ModSlideShow1" prototype is able to receive input of a gesture to move a first content presented on a first display of the plurality of displays as claimed in Claim 5, element (d).

17. The "ModSlideShow1" prototype is able to interpret a direction to move the content from the first display based on the gesture as claimed in Claim 5, element (e).

18. The "ModSlideShow1" prototype is able to specify a second display to which the first content is to be moved based on the gesture and the relative position of the plurality of displays as claimed in Claim 5, element (f).

19. The "ModSlideShow1" is able to present the first content at the second display and a second content at the first display as claimed in Claim 5, element (g).

20. For convenience, Claim 13 of the above application is set out below with elements (h)-(j) identified as follows:

A computer readable medium with instructions for execution by a computer for providing content in a modular presentation system having a plurality of displays, wherein at least two of the plurality of displays are in physical and visual proximity to each other, the instructions comprising:

(h) receiving input of a gesture to move first content presented on a first display;

(i) interpreting a direction to move the content from the first display based on the gesture; and

(j) presenting the first content at the second display and a second content at the first display.

21. The "ModSlideShow1" prototype contained all the elements of Claim 13 as described and claimed in the above U.S. patent application.

22. The "ModSlideShow1" prototype is able to receive input of a gesture to move first content presented on a first display as claimed in Claim 13, element (h).

23. The "ModSlideShow1" prototype is able to interpret a direction to move the content from the first display based on the gesture as claimed in Claim 13, element (i).

24. The "ModSlideShow1" prototype is able to present the first content at the second display and a second content at the first display as claimed in Claim 13,.element (j).

25. A report on the "ModSlideShow: A Slide Presentation System for Peer to Peer Modular Displays with a Gestural Interaction Technique" was prepared on or before April 18, 2002.

26. The report "ModSlideShow: A Slide Presentation System for Peer to Peer Modular Displays with a Gestural Interaction Technique" is shown in a redacted version as Exhibit 'A'.

27. We signed the report on "ModSlideShow: A Slide Presentation System for Peer to Peer Modular Displays with a Gestural Interaction Technique" between the dates of April 18, 2002 and April 22, 2002.

28. A second prototype of the "ModSlideShow" which for convenience we identify as "ModSlideShow2" was reduced to practice on or before April 30, 2002.

29. The "ModSlideShow2" prototype included all the features of the "ModSlideShow1" prototype.

30. The "ModSlideShow2" prototype was implemented with a rear projector and touch screen plasma displays.

31. The "ModSlideShow2" prototype was implemented with laptops connected on a wireless network, where a location sensing system can detect the room location of a laptop.

32. The "ModSlideShow2" prototype integrated a digital compass with a laptop.

33. The "ModSlideShow2" prototype contained all the elements of Claim 1 as described and claimed in the above U.S. patent application.

34. The "ModSlideShow2" prototype comprises a plurality of displays, wherein each display neighbors at least one other display and at least two of the plurality of displays are in visual proximity to each other as claimed in Claim 1, element (a).

35. The "ModSlideShow2" prototype comprises an input device that receives input of a gesture to move a content from a first display of the plurality of displays as claimed in Claim 1, element (b).

36. The "ModSlideShow2" prototype comprises a processor that interprets a direction to move the content from the first display based on the gesture, that specifies a second display to which the content is to be moved, based on the gesture and the position of the plurality of displays and that propagates the content of the first display to the second display, as claimed in Claim 1, element (c).

37. The "ModSlideShow2" prototype contained all the elements of Claim 5 as described and claimed in the above U.S. patent application.

38. The "ModSlideShow2" prototype is able to receive input of a gesture to move a first content presented on a first display of the plurality of displays as claimed in Claim 5, element (d).

39. The "ModSlideShow2" prototype is able to interpret a direction to move the content from the first display based on the gesture as claimed in Claim 5, element (e).

40. The "ModSlideShow2" prototype is able to specify a second display to which the first content is to be moved based on the gesture and the relative position of the plurality of displays as claimed in Claim 5, element (f).

41. The "ModSlideShow2" is able to present the first content at the second display and a second content at the first display as claimed in Claim 5, element (g).

42. The "ModSlideShow2" prototype contained all the elements of Claim 13 as described and claimed in the above U.S. patent application.

43. The "ModSlideShow2" prototype is able to receive input of a gesture to move first content presented on a first display as claimed in Claim 13, element (h).

44. The "ModSlideShow2" prototype is able to interpret a direction to move the content from the first display based on the gesture as claimed in Claim 13, element (i).

45. The "ModSlideShow2" prototype is able to present the first content at the second display and a second content at the first display as claimed in Claim 13, element (j).

46. The "ModSlideShow2" prototype included the command propagation feature.

47. The date of creation of computer files that make up different modules involved in the "ModSlideShow2" prototype is shown in Exhibit B".

48. The date of creation of the ModSlideShow.java file shown in Exhibit B" was April 26, 2002.

49. The date of creation of the GestRecognizer.class file shown in Exhibit B" was April 30, 2002.

50. The "ModSlideShow1" and "ModSlideShow2" prototypes were designed and constructed at the FX Palo Alto Laboratories, Palo Alto, California, United States of America.

51. The "ModSlideShow1" and "ModSlideShow2" prototypes were demonstrated at the FX Palo Alto Laboratories, conference room, in Palo Alto, California, United States of America.

52. The "ModSlideShow1" and "ModSlideShow2" prototypes were demonstrated to an internal FX Palo Alto Laboratories research and development team on or before March 25, 2003.

53. The undersigned, being hereby warned that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. §1001, and that such willful false statements may jeopardize the validity of the application or any resulting registration, declares that the facts set forth in this declaration are true; all statements made of his/her own knowledge are true; and all statements made on information and belief are believed to be true.

2008-07-02	
Date	Patrick Chiu
2008-07-02	Qiong Liu
Date	Qiong Liu
2 JULY 2008	John Steven Boreczky
Date	John Steven Boreczky

FXPAL Invention Proposal FXPAL-IP-02-004

To: Xerox Patent Department
3333 Coyote Hill Road
Palo Alto, CA 94304

Proposal Submitted By:

Patrick Chiu
PAL0042
chiu@pal.xerox.com
(650) 813-7498

Qiong Liu
PAL0063
liu@pal.xerox.com
(650) 813- 6957

John Boreczky
PAL0027
johnb@pal.xerox.com
(650) 813-7585

Manager:

Lynn Wilcox
PAL0020
wilcox@pal.xerox.com
(650) 813-7574

Proposal Title: **ModSlideShow: A Slide Presentation System for Peer-to-Peer Modular Displays with a Gestural Interaction Technique**

Brief Description: It is becoming increasingly common for meeting rooms to be equipped with multiple wall displays. There are also non-fixed displays such as those on laptops or handhelds that are brought into the meeting room by the presenters or other participants. We describe a slide presentation system called *ModSlideShow* for showing and manipulating slides on a heterogeneous set of displays. This system provides a modular structure for the various displays and links them together based on location adjacency and compass orientation; these attributes may be set manually for fixed wall displays and dynamically sensed for mobile displays. A gestural interaction technique enables slides to be pushed or pulled in a particular direction/orientation—with the slides propagating along that direction—and promotes intuitive and fluid interaction. For teleconferences, virtual directions and orientations are supported. Gestures may be performed on a wall display or a mobile computer's display using a touch-panel on top of the display, or alternatively using a pointing device such as a pen, mouse, or laser pointer.

Witnessed and Understood by [signature] *Date* 4/18/02

1. Submitter Signature [signature] *Date* 4-18-02

2. Submitter Signature Qiong Liu *Date* 4-18-02

3. Submitter Signature John Boreczky *Date* 4-22-02

Exhibit A

Description of Invention

Introduction

Presenting slides is an integral part of many meetings. The current standard setup is to use a laptop or meeting room computer to show PowerPoint slides or Web-bases slides on a single video projector. Becoming increasingly common and affordable are wall displays with rear projectors and large plasma displays (see Figure 1). Some of these displays, like modular furniture (see Figure 5), can be rearranged for different types of meetings. In the near future, many meeting rooms will contain a heterogeneous set of various displays including video projectors, plasma panels, displays on tabletops, displays on laptops or notebook computers, and displays on handheld devices. Some of these displays allow the user to interact directly with them, while others merely show the video monitor output of the computer driving it. More generally, a meeting setup may connect a distributed set of meeting rooms and their displays as in a teleconference.

In this IP, we describe a slide presentation system called *ModSlideShow* for showing and manipulating slides on a heterogeneous set of displays. This system provides a modular structure for the various displays and links them together based on location adjacency and orientation. For fixed large wall displays, the location and orientation attributes may be set manually, while for mobile displays (e.g. laptops), the attributes may be dynamically sensed. A number of research location sensing systems have been developed (e.g. see [1], [20], [21]), with ultrasonic systems giving sufficient accuracy (resolution of 9cm). Compass orientation may be detected with a digital compass [4]. More advanced ultrasonic systems [10] claim to do both position and angle with very high precision (position accuracy of 1.5 mm RMS and angular accuracy of 0.05 degree RMS.). For teleconferences, virtual directions and orientations can be defined. The various displays in the ModSlideShow system form a peer-to-peer network that communicate with one another as Web services. The Web services infrastructure is highly suitable for a modular system with heterogeneous components.

In ModSlideShow, the metaphor for the set of presentation slides is a stack of slide images. Each display holds a stack, and the top slide of each stack is shown. A slide is always shown "full screen". Slides can be passed around from display to display by manipulating the stacks of the respective displays.

A gestural interaction technique in ModSlideShow promotes intuitive, fluid, and efficient slide manipulation. We have designed a set of simple gestures that can be made with the finger directly on a touch screen of a plasma panel, with a stylus on a notebook or handheld computer, or with a mouse or other pointing device on a laptop. Alternatively, remote devices such as laser pointers or 3D mice may be used. It is also possible to employ computer vision to detect gestures (e.g. see [6], [24]).

Many of the gesture commands have a direction component. A unique feature of ModSlideShow is command propagation in some direction along a series of adjacent displays. The displays are configured so that each display has only a small number of adjacent displays, typically determined by location and orientation relationships. This configuration is essentially a subgraph of the peer-to-peer network.

Witnessed and Understood by J Foote *Date* 4/18/02

1. Submitter Signature Patrick Chiu *Date* 4-18-02

2. Submitter Signature Qiong Liu *Date* 4/18/02

3. Submitter Signature John Boreczky *Date* 4-22-02

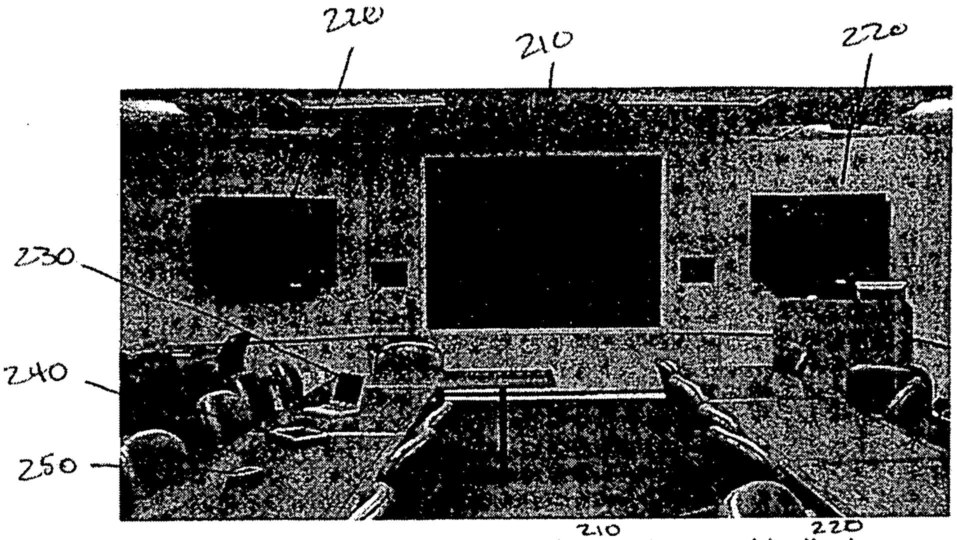


Figure 1. Center display is a rear video projector, the two side displays are plasma displays with touch screens, plus a laptop, pen notebook, and handheld computers

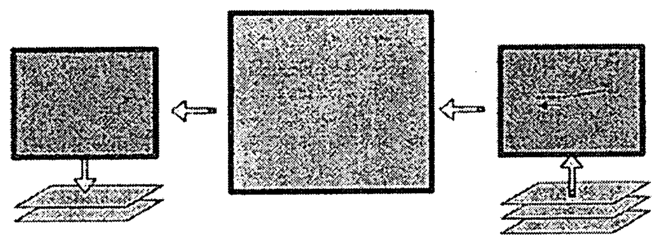


Figure 2. On the right wall display, a line-shaped gesture toward the left pushes its slide to its left (the center display), and command propagation on the center display pushes its slide to the display on its left (the leftmost display)

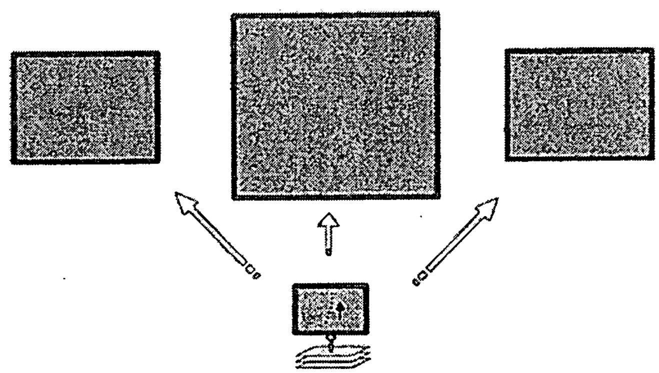


Figure 3. A single gesture can be used to send the next 3 slides in the laptop display's stack to the 3 wall displays

Witnessed and Understood by ________ *Date* 4/18/02

1. Submitter Signature ________ Date 4-18-02

2. Submitter Signature Qiong Liu Date 4/18/02

3. Submitter Signature John Boreczky Date 4-22-02

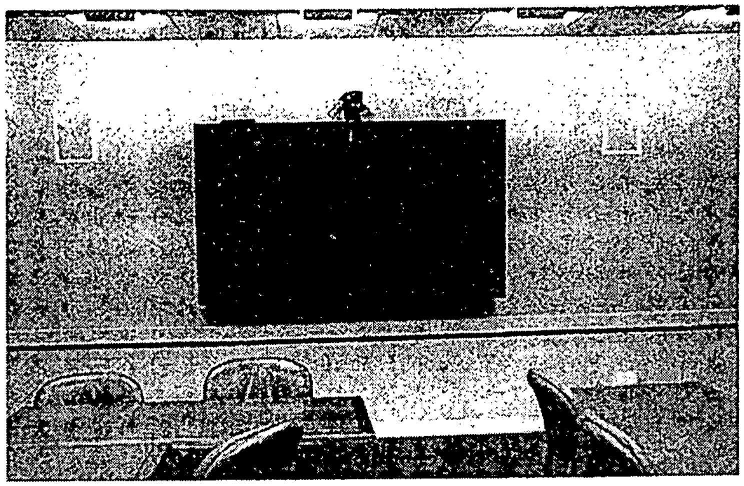


Figure 4. Four plasma displays mounted together on a wall as a mosaic

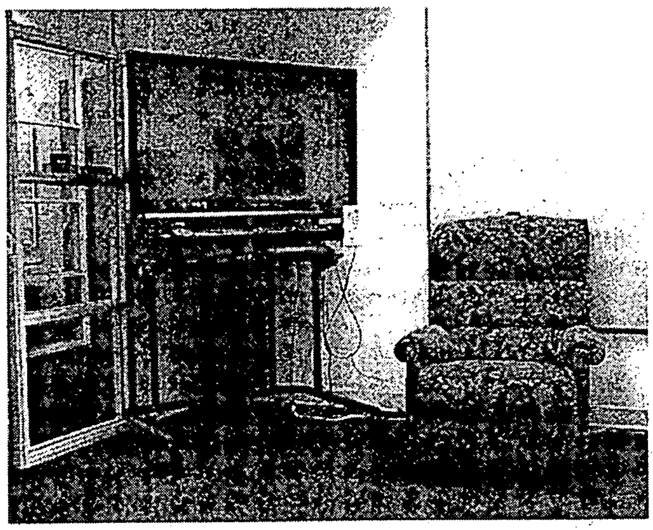


Figure 5. Plasma display with touch screen on a cart

We give two concrete examples to illustrate how ModSlideShow works. Suppose in Figure 1, the presenter starts with a stack of slides on the display on the far right behind the podium. To advance to the next slide, the presenter makes a straight-line gesture, motioning from right to left, on the touch-screen of the display. This is interpreted by the system as a command to push the slide to the display on its right (our left). By allowing commands to propagate, the display on the right sends its top slide to the display on its right. See Figure 2. Upon reaching the last wall display (on our far left), the incoming slide covers the top slide on this display's stack. The result has the desirable effect that the most recent 3 slides are always shown, and each slide operation is accomplished with a single gesture command.

Witnessed and Understood by [signature] *Date* 4/18/02

1. Submitter Signature [signature] Date 4-18-02

2. Submitter Signature Qiong Liu Date 4/18/02

3. Submitter Signature [signature] Date 4-22-02

In our second example, we illustrate how a single gesture can also invoke a transaction with several slides sent to several displays. Suppose in Figure 1, we have a laptop and three wall displays with a stack of slides on the laptop. A straight upward line gesture made on the laptop (interpreted in this case as "put up next bunch of slides") sends the next three slides up to the three displays all at once. See Figure 3. Like the previous example, the user makes far fewer operations than would otherwise be required if each slide were to be moved manually from display to display.

We note that more generally, the techniques described in this IP can be used to move other kinds of content besides presentation slide images around displays. Since what is actually transported is a URL, that URL can be a pointer to any kind of content: video, Web pages, etc.

This IP is organized as follows. In the next two sections we describe more fully the gesture set and the effects of the gesture commands. Then we describe some scenarios of use, technical details, novelty, and related work.

Basic Gesture Set

Basically, a user can give a display a PUSH or PULL command, along with other parameters including the *direction*: R (right), L (left), U (up), D (down), plus the four diagonal directions: UR (up right), UL (up left), DR (down right), DL (down left). Another parameter is *speed*, which is calculated by how rapid the gesture motion is made.

A push gesture is a simple line stroke along with a direction. We specify a line gesture by its direction: G_*direction* (e.g. G_R for a line gesture to the right, G_UL for a diagonal line gesture to the upper left).

A pull gesture is a double-line stroke made by going in one direction and coming back. We specify a double-line gesture by its two successive directions: G_*direction1-direction2* (e.g. G_R-L for double-line gesture right-left, G_UR-DL for a gesture to the upper right and back).

These gestures are simple, intuitive in terms of expressing a direction, easy for users to make and easy for the system to recognize.

Effects of Gesture Commands

After the user inputs a gesture, the system interprets the gesture as a PUSH or PULL command along with parameters for direction and speed. The effect of the gesture depends on whether the display has slides on it, where it is in the modular structure, and its settings for command propagation and command transaction.

At any given time, some of the displays have a set of slides while other displays do not. In the preferred design, slides are shown full-screen without visible UI widgets. Slides may be loaded on to a display by doing the equivalent of a right-click on the screen to bring up a File menu. Each display's set of slides is a stack of images.

A PUSH command moves the slide that is displayed to another display along the direction of the gesture. Usually this is the adjacent display; however, when the speed of the gesture is high, the slide can skim along (and the animated effect is visible) through several displays. The slide is removed from the top of the source display's stack and passed on, and the next slide in the stack is displayed. On the target display, the received slide goes to the top of its stack. A PULL command is the inverse of the PUSH command.

Witnessed and Understood by ______ *Date* 4/18/02

1. Submitter Signature ______ *Date* 4-18-02

2. Submitter Signature Qiong Liu *Date* 4/18/02

3. Submitter Signature ______ *Date* 4-22-02

An option for PUSH/PULL is whether or not to transport a slide or a copy of the slide. For example, a member of the audience may pull a duplicate copy of the slide on a wall display onto his laptop display without disrupting the presentation. Each display can set permissions to specify which other displays can pull a slide or copy of a slide from it.

A display in the modular structure can have up to eight adjacent displays in the eight directions listed in the previous section. The configuration can be set manually (e.g. with fixed wall displays) or dynamically sensed (e.g. with mobile laptop displays). The configuration may also be set for a particular session (e.g. a teleconference). Displays are registered by name and IP address and can be referred to by one another for command transactions.

For each adjacent display, an option may be set to allow commands to propagate. For example, when a display receives a PUSH command to the right along with a slide, it will first execute the command and push its currently displayed slide to the right, and then display the received slide. Typically, it makes sense for shared wall displays to allow propagation and personal mobile displays not to allow it. For shared wall displays, propagation provides a way to keep visible the most recent *N* slides (*N* being the number of wall displays) with minimal effort as the presenter steps through the slides by gesturing on the leftmost (or rightmost) display. On the other hand, for personal mobile displays, propagation is not desirable because having slides shuttle around would be distracting. Both parameters, the direction and speed, are propagated.

A display can also conduct command transactions. A command transaction is a command that executes a set of sub-commands. A useful transaction pushes (or pulls) the next *N* slides on the display's stack (e.g. *N* being the number of wall displays). This enables the user to do a presentation by updating all the wall displays together with a single UP gesture made on her laptop.

Another possibility is to use a transaction to split up a slide image into sections and send each section to a display mosaic (see Figure 4) to construct a larger image.

A more passive feature is enable a display to mirror another display, always obtaining and showing a copy of the slide as it is put up on the original display. This allows audience members and remote sites to obtain a copy of the slide.

For security and access control, displays may also be set to not accept commands from other displays, and more sophisticated schemes including ones similar to group permissions can be employed.

Scenarios of Use

To provide some motivation for the design of the gesture command set and the interaction among the modular displays, we list several scenarios of use. These scenarios are also summarized in Table 1 below.

Witnessed and Understood by [signature] Date 4/18/02

1. Submitter Signature [signature] Date 4-18-02

2. Submitter Signature Qiong Liu Date 4/18/02

3. Submitter Signature John Boreczky Date 4-22-02

	Scenario	# of wall displays	# of mobile displays	# of presenters	Tasks, features
1	Presentation	N > 1	0	1	Step thru recent N slides, "throw" slides
2	Presenter with laptop	N > 1	1 (presenter)	1	Presenter moves laptop around, sense orientation of laptop
3	Discussion, design meeting	N > 1	N > 1 (participants)	N > 1 (using mobile diplays)	Put up slide, share slide, sense location & orientation of mobile displays
4	Presentation with active audience	N > 1	N > 1 (audience)	1	Take snapshot of slide, annotate, put up for Q & A
5	Teleconference	N > 1	N > 1	1 or N > 1	Mirror displays at remote site

Table 1. Scenarios of use

Scenario 1: Fixed Wall Displays Only, with One Presenter

We also assume that at least one of the wall displays has a touch screen for gesture input. In this case, due to the displays being all fixed, the modular structure and adjacency relations among the displays can be manually set (so dynamically determining the location and orientation is not required). In a typical presentation, the presenter creates a set of slides (a directory of JPG or GIF images, which is easily produced by applications such as PowerPoint with its Save As command). The slides are loaded onto one of the displays by using the File Open command under the context menu activated with a right-tap on the display.

In the setup with 3 displays shown in Figure 1, one way to show the slides is for the presenter to stand at the rightmost (from our point of view) display near the podium. As the presenter steps through the slides by making push gestures toward the right, the previously shown slides flow from the rightmost display to the leftmost display, with the most recent 3 slides remaining visible. To go back a slide, the presenter makes a left-right pull gesture and the 3 displays show their previous 3 slides. As the presenter steps through the set of slides, the slides accumulate on the leftmost display. See Figure 2.

Another operation in the Figure 1 setup is to push a slide from the right plasma display to the left plasma display without covering the slide in the large center display. This is done by performing left gesture with a fast motion, which is interpreted as a gesture with high speed. The result is that the slide image skims from the right display to the left display, showing up on the center display momentarily for animation effect.

Scenario 2: Fixed Wall Displays, with One Presenter and his Mobile Display

This is similar to scenario 1, except that the presenter inputs gesture commands on the laptop's display. The laptop sets its adjacent displays either manually by the presenter or automatically using location and orientation sensing technology. Since the speaker faces the audience, the laptop display faces the wall displays behind her. Thus, the orientation of the controlling display is opposite to that in scenario 1. By taking into account the display orientations, the system interprets the gesture directions correctly and the slides end up where the speaker would expect.

Witnessed and Understood by [illegible] Date 4/18/02

1. Submitter Signature [illegible] Date 4-18-02

2. Submitter Signature Qiong Liu Date 4/18/02

3. Submitter Signature John Boreczky Date 4-22-02

A variation is making use of all 3 displays by showing 3 slides at a time. The command transaction option on the presenter's laptop is set to send the next 3 slides to the 3 wall displays. During the presentation, the presenter can make a PUSH UP gesture to invoke this command. See Figure 3.

Scenario 3: Fixed Wall Displays, with Multiple People and their Mobile Displays
This scenario is typical of discussions and design meetings in which a number of people will show a slide from their laptops and handhelds on the large wall display for discussion. There are some interesting possibilities for interaction, which are described in the following.

A user can make a PUSH UP gesture on his laptop to show a slide (or a copy of it) up on a wall display. The display associated with the UP direction may be manually set, or dynamically sensed if the laptop has location and orientation sensors. In the latter case, the user rotates his laptop so that its display and the desired wall display face roughly the same direction. To transport the slide to the display on the right of its UP display, the user makes a diagonal gesture toward the upper right direction.

A user can make a PUSH RIGHT gesture to move the slide from his laptop to his neighbor's laptop on the right. Another way to transfer slides between mobile displays is for one user to PULL a slide to his laptop when another person shows that slide on a wall display.

For sharing slides directly among the laptops, it is natural for two people to move physically near each other. When their laptops are in close proximity, the gesture directions and targets can be accurately interpreted and used to transfer slides. More generally, a menu command can be used to send a slide to any registered local or remote display.

Scenario 4: Fixed Wall Displays, with One Presenter, plus Multiple People and their Mobile Displays
This scenario is like scenario 1 or 2, with audience members also carrying mobile displays. A user in the audience can get a copy of a slide when it is being presented on the wall display. A user in the audience may pull in a copy of a slide from a wall display, annotate it on his pen computer, and put it back up on a wall display for asking question about the highlighted bullet on the slide. In order to avoid too much disruption during a presentation, not all of the displays should be set to allow the audience to send slides to. Also, while it is less likely for the presenter to transport a slide to a member of the audience, the functionality is there.

Scenario 5: Fixed Wall Displays and Mobile Displays in a Teleconference
The general teleconference scenario has similarities to the above scenarios. One difference is that it is desirable to use the mirroring feature and have the slides on the wall displays (or the main large wall display as in the center of Figure 1) shown simultaneously at the remote sites.

Technical Details

Each display runs an instance of the ModSlideShow software. ModSlideShow is a Java application/applet. The instances of ModSlideShow communicate with one another via XML-RPC, which is a Web Services standard. By using the Web Services infrastructure, the system can connect displays in distributed locations for teleconferences.

The slide collections are loaded and reside on Web servers, and the URLs to the slide images are passed around rather than the image files. Since no objects are passed around, a lightweight protocol such as XML-RPC is suitable for this application. When non-fixed displays are supported, a registry for the

Witnessed and Understood by [signature] *Date* 4/18/02

1. Submitter Signature [signature] *Date* 4-18-02

2. Submitter Signature Qiong Liu *Date* 4/18/02

3. Submitter Signature John Boreczky *Date* 4-22-02

discovering the adjacent display is required; this can be a simple listing or handled with a Web services registry. Other issues such as security and transactions can also leverage the technology for handling these in the emerging Web services infrastructure.

Since what is actually transported is a URL, other kinds of content can be transported around the displays besides presentation slide images. The URL can be a pointer to any kind of content: video, Web pages, etc.

Different types of hardware are used to handle gesture input. Fingers can be used on wall displays with touch screens, such as the SMART Board for Plasma Displays [18]. Some displays (e.g. video projectors) may not support gesture input, but they can still show slides and propagate gesture commands from adjacent displays. A stylus is used for gesture input on mobile pen computers such as notebooks and handhelds. Mouse or another pointing device can be used to input gestures on a laptop. There are also laptops with touch screens. Also, remote control devices such as laser pointers or 3D mice may be used with both wall displays and mobile displays.

The gestures used in ModSlideShow are simple line shapes. These are easily recognized by looking at simple geometric features of the gesture path such as the width and height of the bounding box, and the location of the starting and ending points. The speed attribute is calculated by looking at the time difference between the endpoints of the gesture.

Status

Witnessed and Understood by [signature] *Date* 4/18/02

1. Submitter Signature [signature] *Date* 4-18-02

2. Submitter Signature Qiong Liu *Date* 4/18/02

3. Submitter Signature John Boreczky *Date* 4-22-02

Exhibit A

LiteSlideShow [3], a predecessor to ModSlideShow, supports transporting a copy of a slide to a remote display for teleconferences. This has been used with great success for over a year for teleconferences between Palo Alto and Japan.

An early prototype ModSlideShow has been implemented for the FXPAL conference room with a rear projector and touch-screen plasma displays (see Figures 1 and 2). This prototype also works with laptops hooked up on the wireless network. A location sensing system that can detect the room location of a laptop has been deployed (UCLA system [1]), a finer resolution system such as [21] is not yet deployed. We have integrated a digital compass [4] with a laptop (i.e. a mobile display). Emerging Web services standards for security, registry, and transactions have not yet been integrated into ModSlideShow.

References

1. Bolt, R. A. 'Put-That-There': Voice and gesture at the graphics interface, *Proceedings of SIGGRAPH '80*, pp. 262-270.
2. Castro, P., Chiu, P., Muntz, R. A probabilistic room location service for wireless networked environments, Paul Castro. *Proceedings of Ubicomp '01*.
3. Chiu, P., Boreczky, J., Girgensohn, A., Kimber, D. LiteMinutes: An Internet-based system for multimedia meeting minutes. *Proceedings of Tenth International World Wide Web Conference* (2001), pp. 140-149. Available at http://www10.org.
4. The Dinsmore 1490 compass. Dinsmore Sensors. http://www.dinsmoresensors.com/
5. Elrod, S. Liveboard: a large interactive display supporting group meetings, presentations, and remote collaboration. *Proceedings of CHI '92*. ACM, New York, pp.599-607.
6. FXPAL IP-99-024: System for person location and gesture tracking. Inventors: Foote; Jonathan T. / Kimber; Don.
7. FXPAL IP-00-001: LiteMinutes: an internet-based system for multimedia meeting minutes. Inventors: Chiu; Patrick / Kimber; Don / Boreczky; John S. / Girgensohn; Andreas.
8. FXPAL IP-01-018: Systems and Methods for Remote Control of Multiple Displays and Devices. Inventors: Foote; Jonathan T. / Liu; Qiong / Cooper; Matthew / Boreczky; John.
9. Geisler, J. Shuffle, throw or take it! Working efficiently with an interactive wall. *CHI '98 Summary*. ACM Press, pp. 265-266.
10. InterSense IS-900 Precision Motion Tracker. http://www.isense.com/products/prec/is900/IS900.pdf.
11. Li, Kai, *et al.* Early experiences and challenges in building and using a scalable display wall system, *IEEE Computer Graphics and Applications*, vol. 20 (4), pp 671-680, 2000.
12. Moran, T. P., Chiu, P., van Melle, W., Kurtenbach, G. Implicit structures for pen-based systems within a freeform interaction paradigm. *Proceedings of CHI '95*. ACM, New York, pp. 487-494.
13. Myers, B. Using handhelds and PCs together. *Commun. ACM 44*, 11, pp. 34-41.
14. Myers, B.A., Peck, C.H., Nichols, J., Kong, D., Miller, R. Interacting at a distance using semantic snarfing. *Proceedings of Ubicomp '01*.
15. Pedersen, E.R., McCall, K., Moran, T.P., Halasz, F.G. Tivoli: An electronic whiteboard for informal workgroup meetings. *Proceedings of InterCHI '93*. ACM Press, pp. 391-398.
16. Rekimoto, J. Pick-and-Drop: A direct manipulation technique for multiple computer environments. *Proceedings of UIST '97*, pp. 31-39.
17. Rekimoto, J., Ayatsuka, Y. and Hayashi, K., Augment-able reality: Situated communication through digital and physical spaces, *IEEE 2nd Interanational Symposium on Weareble Computer (ISWC '98)*, pp.68-75.
18. SMART Technologies Inc. www.smarttech.com/

Witnessed and Understood by J Foote *Date* 4/18/02

1. Submitter Signature P Chiu *Date* 4-18-02

2. Submitter Signature Qiong Liu *Date* 4/18/02

3. Submitter Signature John Boreczky *Date* 4-22-02

19. Tandler, P., Prante, T., Muller-Tomfelde, C., Streitz, Steinmetz, R. ConnecTables: Dynamic coupling of displays for the flexible creation of shared workspaces. *Proceedings of UIST '01*. ACM, New York, pp.11-20.
20. Want, R. and Schilit, B. (Eds.). Special Issue: Location-Based Computing. *IEEE Computer*, 34, 8 (Aug 2001).
21. Ward, A., Jones, A., Hopper, A. A new location technique for the active office. *IEEE Personal Communications*, Vol. 4, No. 5, October 1997, pp. 42-47.
22. Winograd, T., Guimbretiere, F. Visual instruments for an interactive mural. *Extended Abstracts of CHI '99*. ACM Press, pp. 234-235.
23. Wray, S., Glauert, T., Hopper, A. The Medusa applications environment. *IEEE Multimedia*, Vol. 1, No. 4, Winter 1994, pp. 54-63.
24. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 1997, vol 19, no 7, pp. 780-785.

Others known to have worked on this or a similar invention

See section below on related work.

Related invention proposals, patents, publications or commercial work

The simplest systems for using multiple displays merely show on a set of displays the output of a single computer. Current presentation systems typically use a laptop with a video output signal that shows a copy of the laptop's display on a video projector or a large wall display. In the Unix operating system, a display can be set to view a particular workstation that can be located anywhere on the network. More recently, popular technology such as VNC (Virtual Network Computing) and Microsoft Terminal Services Client allow one computer's display and keyboard/mouse input to control another computer.

Software for application sharing allows the contents of a shared workspace to be viewed on distributed displays, along with supporting user interactions with the shared application. Such software systems include Microsoft NetMeeting, Xerox MeetingBoard and Tivoli [15], and Smart Technologies M-Path [18]. With application sharing, the content shown on one display is similar to that on the other displays.

Smart Technologies SynchronEyes [18] connects a teacher's PC display (which may be projected on a wall) with a number of students' PC displays. The teacher's display can broadcast to all the students' displays, and can also show any student's display to the entire class. Only one slide is shown at a time during a presentation, and the system is not peer-to-peer.

Pebbles SlideShow Commander [13] supports using a PDA to control a PowerPoint presentation on a laptop. The laptop's video output is hooked up in a standard way to the room's wall display. The PDA shows a thumbnail of the slide, and the PDA's buttons are used for moving forward/backward among slides. The presenter has a private view that includes notes to the slides.

A bunch of displays can be tiled together to create a large hi-resolution display. An example is the Princeton Scalable Display Wall [11]. Since the resulting tiled display is intended to behave like a single large display, the spirit of such systems is very different from our loosely coupled modular displays.

Witnessed and Understood by J T Fow *Date* 4/18/02

1. Submitter Signature [signature] Date 4-18-02

2. Submitter Signature Qiong Liu Date 4/18/02

3. Submitter Signature John Boreczky Date 4-22-02

More sophisticated systems that allow separate display units to interact with each other are closer to our work. ConnecTables [19] supports the dynamic coupling of displays for small tables with a display on top. Two tables must be physically moved together in ordered for them to be coupled. Only two displays can be connected at a time, front edge to front edge. The two displays can either form a single surface or remain as two separate surfaces, and the two users can work on computer applications and share data.

Large displays that support gestural input on the screen include the Smart Board and Smart Board for Plasma Displays [18]. Earlier research work on electronic whiteboards includes LiveBoard [5] and its Tivoli software [12]. Tivoli uses gestures for manipulating content on a whiteboard-sized display surface (a copy of the work surface may be shown via application sharing explained above). For larger wall-sized displays, an interaction technique developed at GMD [9] uses gestures to manipulate objects on a single surface; it has a "throw" gesture for moving an object across the display, and a "shuffle" gesture for moving a single object a by a distance equal to its length. Another project at Stanford [22] explores using gestures made by a laser pointer on a single large continuous display.

Several interaction techniques were invented at Sony for moving data between displays. Pick-and-drop [16] is an interaction technique for moving data between displays by directly "picking" up an object from one display with a pointing device and "dropping" it on another display. Picking and dropping involves touching the displays with a pen, so location and orientation of the displays are irrelevant. Pick-and-drop affects only the displays touched by the pen, and actions are not propagated to other displays. Another technique called hyperdragging [17] extends the reach of a cursor seamlessly from a portable laptop's display to the displays (projected) on the tables and walls in an augmented environment, and objects can be dragged and dropped among different displays and computers. With hyperdragging, the displays are modeled as one continuous surface for the cursor to navigate in. Location and orientation of tagged laptops are detected with video cameras and are used to determine the edges of the laptop's display to determine the placement of the cursor as it jumps to another display.

While our main scenario is characterized by direct manipulation, with the user touching the display's screen or using the display computer's input device to perform push and pull gestures on the presentation slides, it is also possible to make the gestures with a separate remote control device or laser pointer. In FXPAL-IP-01-018 [8], a hand-held remote control device is proposed for controlling multiple audio, video and other displays and peripherals. It is interfaced with a media switching system and the user can "select" and "paste" among the desired displays or output devices. In one of the embodiments, a six-degree-of-freedom pointer device that has location and orientation tracking is employed. A technique for handheld computers at CMU called "semantic snarfing" [13] enables the meaning of a screen's contents, rather than a screenshot, to be copied to the handheld (e.g. a snarfed menu is not just an image of the menu but a functioning menu). This mechanism allows cutting and pasting among displays. Some earlier work at MIT, Spatial Data-Management System (SDMS) [1], uses wall display and two monitors next to the armrests of a chair. One of the monitors shows an overview of the data; the user can select some small portion of data and the selected data is displayed in a more detail view on the large screen. SDMS also explored using speech recognition with a position-orientation Polhemus device to enable a user to draw a shape on the wall display (e.g. by pointing and saying "Create a blue circle there.").

Infrastructure for peer-to-peer software modules for controlling multimedia devices has been explored in the Medusa system [23]. The hyperdragging system [17] uses Java and RMI (remote method invocation) to manage data between computers (which are modeled as one continuous interaction or surface). Our modular peer-to-peer software model is simpler and more easily integrate set of heterogeneous displays because ours is based on the Web Services infrastructure standards.

Witnessed and Understood by [signature] *Date* 4/18/02

1. Submitter Signature [signature] *Date* 4-18-02

2. Submitter Signature Qiong Liu *Date* 4/18/02

3. Submitter Signature John Boreczky *Date* 4-22-02

Has invention been built, made, run, or tested?

See Status section.

Is the invention used in a current product(s) or planned for use in a future product(s)?

ModSlideShow technology will likely be used and/or licensed in FX Smart Spaces products.

Dates of any previous or planned future disclosures external to Xerox

UIST'02, (November 2002), or later in CHI '03 (April 2003), precise dates for these conferences have not been finalized.

Source of outside funding

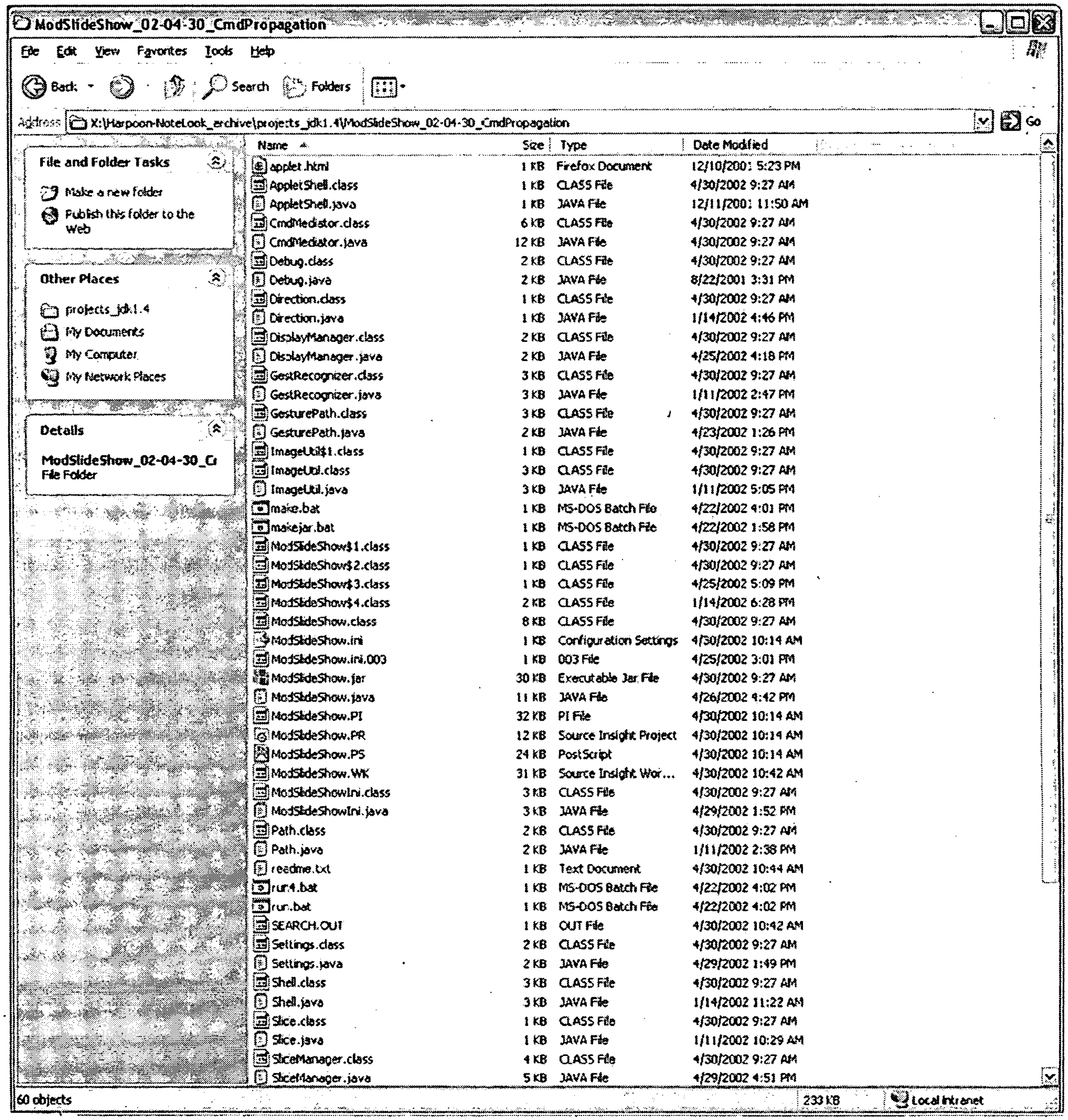
None.

Witnessed and Understood by [signature] *Date* 4/18/02

1. Submitter Signature [signature] *Date* 4-18-02

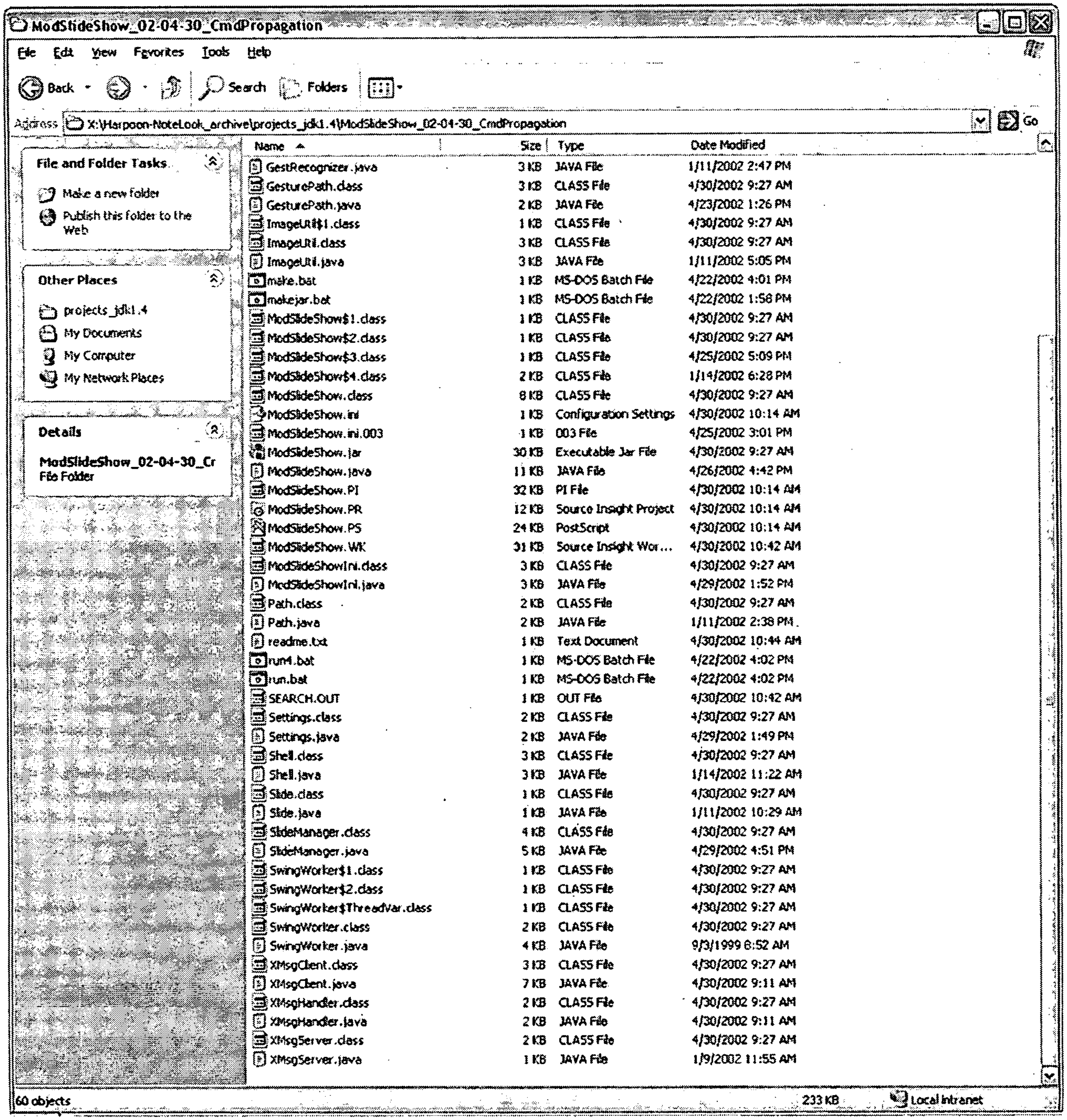
2. Submitter Signature Qiong Liu *Date* 4/18/02

3. Submitter Signature John Boreczky *Date* 4-22-02



Name	Size	Type	Date Modified
applet.html	1 KB	Firefox Document	12/10/2001 5:23 PM
AppletShell.class	1 KB	CLASS File	4/30/2002 9:27 AM
AppletShell.java	1 KB	JAVA File	12/11/2001 11:50 AM
CmdMediator.class	6 KB	CLASS File	4/30/2002 9:27 AM
CmdMediator.java	12 KB	JAVA File	4/30/2002 9:27 AM
Debug.class	2 KB	CLASS File	4/30/2002 9:27 AM
Debug.java	2 KB	JAVA File	8/22/2001 3:31 PM
Direction.class	1 KB	CLASS File	4/30/2002 9:27 AM
Direction.java	1 KB	JAVA File	1/14/2002 4:46 PM
DisplayManager.class	2 KB	CLASS File	4/30/2002 9:27 AM
DisplayManager.java	2 KB	JAVA File	4/25/2002 4:18 PM
GestRecognizer.class	3 KB	CLASS File	4/30/2002 9:27 AM
GestRecognizer.java	3 KB	JAVA File	1/11/2002 2:47 PM
GesturePath.class	3 KB	CLASS File	4/30/2002 9:27 AM
GesturePath.java	2 KB	JAVA File	4/23/2002 1:26 PM
ImageUtil$1.class	1 KB	CLASS File	4/30/2002 9:27 AM
ImageUtil.class	3 KB	CLASS File	4/30/2002 9:27 AM
ImageUtil.java	3 KB	JAVA File	1/11/2002 5:05 PM
make.bat	1 KB	MS-DOS Batch File	4/22/2002 4:01 PM
makejar.bat	1 KB	MS-DOS Batch File	4/22/2002 1:58 PM
ModSlideShow$1.class	1 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShow$2.class	1 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShow$3.class	1 KB	CLASS File	4/25/2002 5:09 PM
ModSlideShow$4.class	2 KB	CLASS File	1/14/2002 6:28 PM
ModSlideShow.class	8 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShow.ini	1 KB	Configuration Settings	4/30/2002 10:14 AM
ModSlideShow.ini.003	1 KB	003 File	4/25/2002 3:01 PM
ModSlideShow.jar	30 KB	Executable Jar File	4/30/2002 9:27 AM
ModSlideShow.java	11 KB	JAVA File	4/26/2002 4:42 PM
ModSlideShow.PI	32 KB	PI File	4/30/2002 10:14 AM
ModSlideShow.PR	12 KB	Source Insight Project	4/30/2002 10:14 AM
ModSlideShow.PS	24 KB	PostScript	4/30/2002 10:14 AM
ModSlideShow.WK	31 KB	Source Insight Wor...	4/30/2002 10:42 AM
ModSlideShowIni.class	3 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShowIni.java	3 KB	JAVA File	4/29/2002 1:52 PM
Path.class	2 KB	CLASS File	4/30/2002 9:27 AM
Path.java	2 KB	JAVA File	1/11/2002 2:38 PM
readme.txt	1 KB	Text Document	4/30/2002 10:44 AM
run4.bat	1 KB	MS-DOS Batch File	4/22/2002 4:02 PM
run.bat	1 KB	MS-DOS Batch File	4/22/2002 4:02 PM
SEARCH.OUT	1 KB	OUT File	4/30/2002 10:42 AM
Settings.class	2 KB	CLASS File	4/30/2002 9:27 AM
Settings.java	2 KB	JAVA File	4/29/2002 1:49 PM
Shell.class	3 KB	CLASS File	4/30/2002 9:27 AM
Shell.java	3 KB	JAVA File	1/14/2002 11:22 AM
Slice.class	1 KB	CLASS File	4/30/2002 9:27 AM
Slice.java	1 KB	JAVA File	1/11/2002 10:29 AM
SliceManager.class	4 KB	CLASS File	4/30/2002 9:27 AM
SliceManager.java	5 KB	JAVA File	4/29/2002 4:51 PM

Exhibit B



Name	Size	Type	Date Modified
GestRecognizer.java	3 KB	JAVA File	1/11/2002 2:47 PM
GesturePath.class	3 KB	CLASS File	4/30/2002 9:27 AM
GesturePath.java	2 KB	JAVA File	4/23/2002 1:26 PM
ImageUtil$1.class	1 KB	CLASS File	4/30/2002 9:27 AM
ImageUtil.class	3 KB	CLASS File	4/30/2002 9:27 AM
ImageUtil.java	3 KB	JAVA File	1/11/2002 5:05 PM
make.bat	1 KB	MS-DOS Batch File	4/22/2002 4:01 PM
makejar.bat	1 KB	MS-DOS Batch File	4/22/2002 1:58 PM
ModSlideShow$1.class	1 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShow$2.class	1 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShow$3.class	1 KB	CLASS File	4/25/2002 5:09 PM
ModSlideShow$4.class	2 KB	CLASS File	1/14/2002 6:28 PM
ModSlideShow.class	8 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShow.ini	1 KB	Configuration Settings	4/30/2002 10:14 AM
ModSlideShow.ini.003	1 KB	003 File	4/25/2002 3:01 PM
ModSlideShow.jar	30 KB	Executable Jar File	4/30/2002 9:27 AM
ModSlideShow.java	11 KB	JAVA File	4/26/2002 4:42 PM
ModSlideShow.PI	32 KB	PI File	4/30/2002 10:14 AM
ModSlideShow.PR	12 KB	Source Insight Project	4/30/2002 10:14 AM
ModSlideShow.PS	24 KB	PostScript	4/30/2002 10:14 AM
ModSlideShow.WK	31 KB	Source Insight Wor…	4/30/2002 10:42 AM
ModSlideShowIni.class	3 KB	CLASS File	4/30/2002 9:27 AM
ModSlideShowIni.java	3 KB	JAVA File	4/29/2002 1:52 PM
Path.class	2 KB	CLASS File	4/30/2002 9:27 AM
Path.java	2 KB	JAVA File	1/11/2002 2:38 PM
readme.txt	1 KB	Text Document	4/30/2002 10:44 AM
run4.bat	1 KB	MS-DOS Batch File	4/22/2002 4:02 PM
run.bat	1 KB	MS-DOS Batch File	4/22/2002 4:02 PM
SEARCH.OUT	1 KB	OUT File	4/30/2002 10:42 AM
Settings.class	2 KB	CLASS File	4/30/2002 9:27 AM
Settings.java	2 KB	JAVA File	4/29/2002 1:49 PM
Shell.class	3 KB	CLASS File	4/30/2002 9:27 AM
Shell.java	3 KB	JAVA File	1/14/2002 11:22 AM
Slide.class	1 KB	CLASS File	4/30/2002 9:27 AM
Slide.java	1 KB	JAVA File	1/11/2002 10:29 AM
SlideManager.class	4 KB	CLASS File	4/30/2002 9:27 AM
SlideManager.java	5 KB	JAVA File	4/29/2002 4:51 PM
SwingWorker$1.class	1 KB	CLASS File	4/30/2002 9:27 AM
SwingWorker$2.class	1 KB	CLASS File	4/30/2002 9:27 AM
SwingWorker$ThreadVar.class	1 KB	CLASS File	4/30/2002 9:27 AM
SwingWorker.class	2 KB	CLASS File	4/30/2002 9:27 AM
SwingWorker.java	4 KB	JAVA File	9/3/1999 6:52 AM
XMsgClient.class	3 KB	CLASS File	4/30/2002 9:27 AM
XMsgClient.java	7 KB	JAVA File	4/30/2002 9:11 AM
XMsgHandler.class	2 KB	CLASS File	4/30/2002 9:27 AM
XMsgHandler.java	2 KB	JAVA File	4/30/2002 9:11 AM
XMsgServer.class	2 KB	CLASS File	4/30/2002 9:27 AM
XMsgServer.java	1 KB	JAVA File	1/9/2002 11:55 AM

Exhibit B